(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
3 May 2001 (03.05.2001)

PCT

(10) International Publication Number
WO 01/31920 A1

(51) International Patent Classification[7]: H04N 7/16

(21) International Application Number: PCT/US99/24745

(22) International Filing Date: 22 October 1999 (22.10.1999)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant *(for all designated States except US)*: GENERAL INSTRUMENT CORPORATION [US/US]; 101 Tournament Drive, Horsham, PA 19044 (US).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **BOOTH, Robert, Charles** [US/US]; 1700 Rockcress Drive, Jamison, PA 18929 (US). **TAVOLETTI, Donald** [US/US]; 2268 Ridge View Drive, Warrington, PA 18976 (US). **BATES, Thomas, E., IV.** [US/US]; 115 Tanyard Road, Richboro, PA 18954 (US). **DEL SORDO, Chris** [US/US]; 229 Heatherfield Drive, Souderton, PA 18964 (US). **ERINOFF, Mark, A.** [US/US]; 195 Middle Park Drive, Souderton, PA 18964 (US). **DIFIGLIA, Michael** [US/US]; 2043 Oakdale Avenue, Glenside, PA 19038 (US).

(74) Agent: **LIPSITZ, Barry, R.**; Building No. 8, 755 Main Street, Monroe, CT 06468 (US).

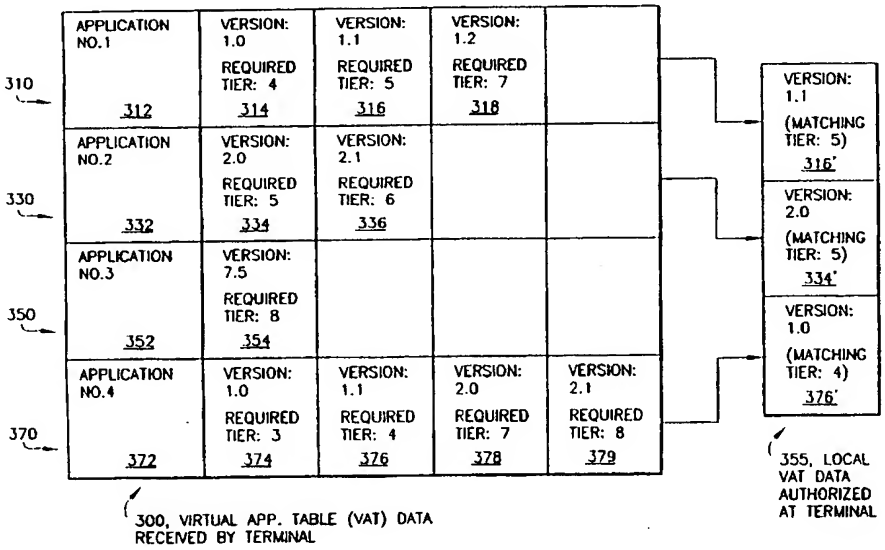
(81) Designated States *(national)*: AE, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:
— *With international search report.*
— *With amended claims.*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: METHOD AND APPARATUS FOR MANAGING MULTIPLE APPLICATIONS IN LARGE SCALE NETWORKS



(57) Abstract: A digital network manages and utilizes multiple applications in consumer terminals (150), such as set-top boxes. Terminals are authorized individually to use one or more of a number of available applications, such as e-mail, video on demand, stock ticker, or a web browser. A download message from a headend (115) authorizes each terminal (150) to download and use specific applications by building its own "local" Virtual Application Table (VAT) (355). Different terminals can be authorized to recover and use different versions (314, 316, 318; 334, 336; 354; 374, 376, 378, 379) of the same application (312; 332; 352; 372) which, e.g., provide enhanced features over the basic application, or provide a test version for trouble-shooting or test marketing.

WO 01/31920 A1

METHOD AND APPARATUS FOR MANAGING MULTIPLE APPLICATIONS IN LARGE SCALE NETWORKS

BACKGROUND OF THE INVENTION

The present invention relates to a method and apparatus that allows a digital network to manage and utilize multiple applications in consumer terminals (e.g., set-top boxes). The applications can be provided in different service tiers on a fee basis. The invention provides a "Multiple Application Management (MAM)" feature by defining mechanisms, messages and data structures.

The communication of data via digital networks, including broadband communication networks such as cable and satellite television networks, has become increasingly popular. Such networks allow consumers and others to receive high quality video and audio programming services. Moreover, commonly an application such as an Electronic Program Guide (EPG) which lists the available programming services has been made available.

With the increasing integration of computer networks such as the Internet, telephony networks, and broadband distribution networks, many opportunities arise for providing new types of applications, such as electronic program guides, Internet browsers, video on demand, audio on demand, mail services (e.g., text e-mail, voice mail, audio mail, and/or video mail), telephony services, stock prices, weather data, travel

information, games, gambling, banking, shopping, voting, and others.

However, currently there is a lack of capability within broadband digital terminal networks to efficiently support more than one software application. As mentioned, typically this single software application is an electronic program guide. Thus, existing digital terminals cannot support additional software applications that can enhance the user's experience and increase revenue for the service provider.

Accordingly, it would be desirable to provide the capability for a digital terminal to download more than one application, and to manage the resources used by different applications. The system should manage the authorization and enabling of the different applications in different digital terminals in a terminal population. The system should inform the user as to what applications are authorized and available for use.

Generally, the system should:

1. allow a digital terminal to support multiple applications;

2. inform the terminal and end user, e.g., via an on-screen menu, which applications are authorized in a given terminal, and allow the user to select (e.g., "launch") an application;

3. indicate to a digital terminal if an application has special features (e.g., such as built-in e-mail), and authorize these features in the digital terminal;

4. indicate that a specific programming source or channel should be tuned to before an application is launched;

5. effectively manage volatile and non-volatile memory (terminal resources) used by an application;

6. authorize an application on a terminal via a Billing System (e.g., at a headend), and provide the different applications on a fee basis, including the provision of different service tiers;

7. provide conditional downloading of applications while avoiding unnecessary expense in terms of security processing;

8. provide backward compatibility with existing terminals in the network (e.g., operator's plant) to allow the terminals to continue to operate without any detrimental side effects; and

9. provide updated and test versions of applications to specific terminals, and manage the enabling thereof at the terminals.

The present invention provides a system having the above and other advantages.

SUMMARY OF THE INVENTION

The present invention relates to a method and apparatus for allowing a digital network to manage and utilize multiple applications in consumer terminals.

Significant features of the present invention include:

1. Use of a digital message (such as a Virtual Object message encapsulated within an MPEG message), transmitted across any network (such as a broadband cable network), to deliver the following to a digital consumer terminal:

1-A. Software application specific information (such as a Virtual Application Table, or VAT), including, but not limited to:

i) Application authorization requirements used to authorize the download of an application to a digital terminal;

ii) Index or reference to an authorized application (such as on a broadband network multiplex or on the Internet);

iii) Application authorization requirements used to authorize the execution of (or enabling of) an application in a digital terminal;

iv) Application feature authorization requirements used to authorize such features on a digital terminal (e.g., built-in e-mail, video-on-demand, or web browsing capabilities associated with an application such as an electronic program guide);

v) Application-specific commands and operations to be executed prior to the download and/or launch of

an application, such as tuning to a specific channel (or channels, if multiple tuners are available);

vi) Application menu data used by a digital terminal for creating a menu of authorized applications available for selection by an end user, such as text data describing one or more authorized applications (i.e., the names of the applications); and

1-B. Initialization and configuration information allowing:

i) A digital terminal to be configured for multiple applications;

ii) A digital terminal to be allocated a specified amount of volatile memory for the download of multiple applications; and

iii) A digital consumer terminal to receive the software.

2. The invention also provides dynamic creation of user application selection menu(s) containing a list of applications which are currently authorized for a digital terminal.

3. The invention also provides removal of application data in non-volatile or volatile memory based upon the current authorization state of application versions.

4. The invention also provides authorization of the following via a billing system interface:

A. Access to broadcast and/or interactive data servers allowing for access to data objects or data services, such as software applications and associated application features.

In accordance with the present invention, a method for managing multiple applications in a digital network having a headend that broadcasts programming services to a terminal population via a communication channel, includes the step of: communicating configuration data from the headend to the terminals in the terminal population via the communication channel to configure the terminals to receive application data and control data. The application data is provided for a plurality of applications, and defines an identifier, a version, and a required authorization state for each of the applications.

The control data defines respective authorization states (or tiers) for the terminals. The application data and control data are communicated from the headend to the terminals via the communication channel to enable the terminals to download and access the versions of the applications for which the required authorization state thereof corresponds to the terminal's authorization state.

The terminal automatically downloads any new version of an application for which it is authorized to replace the old version. Similarly, if the terminal's authorization state is upgraded, e.g., upon payment of additional fees by the subscriber, additional applications are downloaded. If the terminal's authorization state is downgraded, applications already stored but no longer authorized are deleted.

A corresponding method for managing multiple applications at a terminal is disclosed.

Corresponding apparatuses are also disclosed.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 illustrates an overview of a digital network for providing multiple application management in accordance with the present invention.

FIG. 2 illustrates a digital terminal with a multiple application management capability in accordance with the present invention.

FIG. 3 illustrates the creation of a local virtual application table (VAT) for a terminal based on the terminal's authorized tiers in accordance with the present invention.

FIG. 4 illustrates an example dynamically-created menu of the available applications at a terminal in accordance with the present invention.

FIG. 5 illustrates a method for providing multiple application management in a digital network in accordance with the present invention.

DETAILED DESCRIPTION OF THE INVENTION

The present invention relates to a method and apparatus for allowing a digital network to manage and utilize multiple applications in consumer terminals.

FIG. 1 illustrates an overview of a digital network for providing multiple application management in accordance with the present invention. A business system 105, which may be located at, or otherwise be in communication with, the headend 115 of a network such as a cable or satellite television network, manages the billing and authorization of applications for each specific terminal in a network.

Users of the network can make arrangements to receive authorizations for the applications using conventional techniques, e.g., by phoning an operator and authorizing a credit card payment, or by use of an upstream communication path on the network, if available. For example, a user may request an authorization for an e-mail application, assuming the terminal has the capability to access a network such as the Internet. Moreover, the user may have the option of requesting a basic or an enhanced e-mail capability for different fees.

Thus, a virtual application can be viewed as a "service" from the perspective of the Business System 105.

Moreover, the network operator has the capability to authorize specific terminals to receive an application without a user request, e.g., as a promotion, or as part of a package deal when other

programming services are ordered, or some other goal is reached, such as the user purchasing a certain dollar value of video-on-demand programs.

The business system 105 can be implemented with a computer and known record-keeping and billing procedures.

The business system 105 communicates with a controller 120, which communicates with a download server 110. The download server 110 transmits the application data via an interface 130, and physical network and intermediate equipment 140 to a terminal 150. Note that the example terminal 150 is assumed to be part of a large terminal population. The application data may be broadcast to all terminals, but preferably can only be recovered by the terminals based on control data from the controller 120.

Alternatively, or in addition, control data can be provided to the terminal 150 by other means, such as locally using a smart card, or at the time of installation or manufacture of the terminal. However, the provision of control data via a controller that is under the direct control of a headend 115 is believed to provide the greatest flexibility since updated control data can be transmitted immediately to the terminal 150. Moreover, known decoder addressing and conditional access techniques can be used to deliver specific control data to specific terminals or groups of terminals. For example, the control data can be encrypted under a key that has been assigned to the specific terminal.

The controller 120 thus configures and authorizes the terminals under the control of the Business System 105.

Programming services, e.g., conventional television programs, or other video, audio or other data, is provided by a programming services function 125.

The application and control data can be encapsulated in transport packets, for example, such as MPEG-2 packets, using known techniques. The application and control data can be carried in-band, with the programming services, or out-of-band, apart from the programming services. Also, the application data can be sent via any reliable transport mechanism, for example via TCP/IP.

The physical network and intermediate equipment 140 may include cable and/or optical fiber, as well as required switches, amplifiers and other conventional components.

FIG. 2 illustrates a digital terminal with multiple application management capability in accordance with the present invention. Like-numbered elements correspond to one another in the figures. The terminal 150 receives MPEG messages (packets), such as an example packet 205, from a communication channel. Use of MPEG packets is discussed herein only as an example. Any digital data transport protocol may be used.

An MPEG packet processor and packet identifier (PID) filter 210 processes the packet 205 to recover the control data from the controller 120 of FIG. 1,

which is provided to a security processor 250 and a Multiple Application Manager (MAM) 240. The MAM 240 and other terminal functions can be implemented using any known software, firmware and/or hardware techniques.

The control data, including authorization state data, can be stored at a memory associated with the terminal 150. The packet processor 210 also recovers the application data and forwards it to a downloader 230. The downloader 230 has an associated memory for storing the downloaded application data, including the applications themselves, such as code objects. "Downloading" refers to recovering and storing. The downloader 230 also receives a *"Tune Download Channel"* message that commands it to download particular applications, and/or particular versions of the same application from a specific channel. The channel may be identified by a PID in a known manner.

The packet processor 210 may also recover conventional programming services for decoding, e.g., at an MPEG video decoder 215, and display on a display 200.

The MAM 240 can output data to the display 200 for launching the applications, such as e-mail or web browser, stock ticker, or the like, separately or together with data from a programming service.

The MAM 240 can also output data to the display 200 for providing an on-screen menu of available applications (see FIG. 4). A user may interact with the menu via a user command processor 255, e.g., which

receives input signals from a keyboard, infra-red remote control of the like.

The security processor 250, a Local Virtual Application Table (VAT) memory 260, a Home VAT data memory 265, and a message router/filter 225 communicate with the MAM 240. Home VAT data refers to common VAT table that is downloaded to all terminals, while local VAT data refers to data that is used by each terminal, and is derived as a subset of the home VAT data based on the terminal's authorization state. Essentially, the local VAT data designates the latest version of each application that a terminal is authorized to download and access. The local VAT data can therefore be different for different terminals.

The message filter/router 225 sends data such as control data and authorization data, including the home VAT data and Entitlement Management Messages (EMMs) to the MAM 240, while the application code (software) is sent to the downloader 230.

With the present invention, control data is used to authorize terminals to acquire multiple applications, and to enable the applications for use within the terminal.

The MAM 240 can be implemented by using new messages in the terminal 150, as well as some existing messages that are modified and/or interpreted differently.

The MAM 240 receives and processes these messages, and uses the security processor 250 to determine which of the multiple applications is authorized for acquisition and enabling at the terminal.

Virtual applications are applications that can be identified, downloaded, and enabled under the control of the MAM 240. The virtual applications can be transported to the terminal 150 in download messages. However, the applications could downloaded via other means, such as via HTTP.

Virtual Application configuration messages and Virtual Object messages, which contain Virtual Application Tables, are examples of messages which can be used in a digital network, such as the one depicted in FIG. 1, for managing multiple applications, and for configuring a terminal for multiple application management.

The data structures and information contained in the messages also provide authorization requirements needed by a terminal for downloading an application, and for enabling and executing an application or any special characteristics that may be associated with the application.

In addition, via other messages sent by the controller, the terminal receives authorization rights for an application or for any special characteristics associated with an application.

For example, EMMs sent from the controller can authorize a terminal for an application or its characteristics. This is done in the same manner that a terminal is authorized for a video service.

Using the authorization requirements and the authorization rights, the MAM 240 uses the security processor module 250 within the terminal 150 to determine the authorization state, or other special

characteristics, of any given version of an application.

In one possible implementation, the MAM 240 maintains the required authorization state of a virtual application in non-volatile memory within the terminal 150.

The authorization state of an application determines if an application can be downloaded by the downloader 230 at the terminal 150, i.e., whether an application (or a specific version of the application) is preserved in, purged from, or deleted from, the downloader's memory.

Also, the authorization state of an application may determine whether or not specified resources can be pre-allocated for the application in the terminal, such as the amount of volatile and non-volatile memory.

The messages from the controller 120 also provide the terminals with additional information pertaining to the applications, such as an index or reference to an application code object on a network.

The messages also provide information regarding the specific channel or channels which the terminal 150 may tune to for acquiring video, audio, and/or data content associated with the applications.

The VAT data may be stored in non-volatile flash memory, battery-backed SRAM, a hard drive if available, or any other non-volatile memory available in the terminal. Alternatively, the VAT data may be stored in volatile memory, in which case it is simply re-acquired from the network each time the terminal is powered on.

Moreover, the VAT data may be sent to the terminal

150 from the controller 120 on a cyclic basis, e.g., every twenty seconds. However, this time frame can be adjusted based upon specific network configurations and demands.

A virtual application can be configured and enabled as follows. The terminal 150 may optionally receive a configuration message ("Virtual Application Config") that informs it that it is configured in a MAM state. When so configured, the terminal can receive a "virtual object message", which provides the home VAT to the terminal. The terminal 150 derives it's own local VAT 260 based on the received home VAT data 265 and the received control data, which sets the terminal's authorized tier(s) (e.g., authorization state).

The MAM 240 can maintain the information from the Virtual Application Config. and the Virtual Object messages in non-volatile memory. The information would thus be preserved through any warm resets of the terminal. A warm reset causes volatile memory, such as DRAM to be cleared/reset. This may be caused by unplugging the terminal from its power supply, for example.

The MAM 240 communicates with the security processor 250 to check the required authorization tiers for applications, which are specified in the received VAT data. The authorization state information is typically maintained in non-volatile memory.

The downloader 230 maintains a directory of the versions of the code objects that are already stored.

If the *Tune Download Channel* message for a virtual application is received by the downloader 230, the downloader 230 checks its object directory to determine if the version of the code object specified in the message is already present. If the code object (e.g., application) is not already present, the downloader 230 will check with the MAM 240 to determine if the version of the application is authorized for downloading.

If the MAM 240 informs the downloader 230 that a specific virtual application version is authorized, the downloader 230 tunes to a download channel for the application and attempts to acquire the specified application version. After acquiring a virtual application, the downloader 230 de-tunes from the download channel.

The applications can be assigned a default "disabled" status when first recovered. In this case, the MAM 240 also informs the downloader 230 whether to enable the applications or to leave them disabled. Alternatively, the applications can be automatically enabled when they are recovered by the downloader 230.

In one possible implementation, only one application is enabled at any given time. This may be the case when current applications do not gracefully share resources, such as memory, queues, and so forth, so the MAM 240 has to disable an application to reacquire these resources for use by another application. Preferably, the terminal has the capability of enabling more than one application at a time. To accomplish this, a second set of states, or

modes of operation, can be added for applications that are enabled. This set would consist of foreground and background modes. In this type of implementation, only one application will ever be the "foreground" application at any time.

The MAM can also dynamically prepare an on-screen menu based upon the authorized virtual applications, as discussed further in connection with FIG. 4.

Moreover, the MAM 240 can tune, if specified, to a channel (or channels) that are associated with a virtual application prior to, concurrent with, or after launching the virtual application. For example, the virtual application may comprise a banner of sports scores, in which case the MAM 240 can cause a specific sports-oriented programming service (e.g., ESPN(tm)) to be tuned and displayed. Conversely, the application of sports scores may be automatically launched when the programming service is tuned.

The MAM 240 can re-check the required authorizations of all virtual applications if the terminal 150 receives a new VAT, receives a change in its existing VAT, or receives new authorization rights, e.g., via an EMM. The VAT data and control messages can be transmitted to the terminals on an on-going basis, at regular intervals, or only at specific times.

Optionally, each VAT may have revision data, such as a sequence number, that changes whenever application versions are added or removed. The MAM 240 is alerted by the revision data to re-check its authorizations and modify its local VAT, if necessary. Or, the MAM 240

may simply recheck its authorizations periodically or based on some other criteria.

Efficient management of a terminal's memory resources is also an important part of the present invention. Accordingly, the MAM 240 can provide control signals to the downloader 230, based upon the existence of and/or the authorization state of a virtual application, which determine whether or not the downloader 230 should remove or maintain the code objects related to the virtual application(s) that are stored in the downloader's memory. Generally, lower versions of an application (authorized or unauthorized) that are being replaced should be deleted when the new highest authorized version is downloaded.

Advantageously, it is possible to upgrade only a subset of the terminals in a terminal population by providing a decoder conditional, *"configured_for_MAM"*, that determines whether each terminal will acquire a VAT and become MAM enabled, and tune to a download channel to acquire a virtual application. Thus, terminals that have not been upgraded with MAM-capable firmware platform code can continue to operate without any detrimental side effects caused by the innovations involved with MAM. On the other hand, *"configured_for_MAM"* can be set to allow progressive upgrading of a terminal population.

The *"Virtual Application Config"* message (e.g., "configuration data") is used to configure or de-configure a terminal for MAM, and to provide MAM configuration settings to a terminal. Information

derived from the *Virtual Application Config* message is typically stored by the terminal in non-volatile memory (e.g., via the MAM 240) to preserve it through (warm) resets of the terminal.

The *Virtual Application Config* message can include the following significant fields in an example syntax:

"config_for_multi_apps", when set to "yes", configures a terminal for MAM capability. The terminal is then considered to be in a *configured_for_MAM* state, and is able to receive other messages which have the *configured_for_MAM* decoder condition in the message preamble. If this field is cleared to "no", the terminal will no longer be *configured_for_MAM*, nor enabled for MAM;

"home_VAT_ID" identifies a VAT which is used as a terminal's default VAT (*"home_VAT"*);

"default_application_ID" identifies an application which will be the default virtual application for a terminal. This ID correlates to the *object_application_ID* of a virtual application in the *home_VAT*; and

"volatile_memory_config" specifies the number of bytes of volatile memory that the terminal allocates and make available for the download of virtual applications other than the default virtual application.

Furthermore, the present invention can use a message type known as a "Virtual Object message", e.g., to deliver a VAT to a terminal. Moreover, this message

can be carried in a network stream (an MPEG standard that denotes any data delivered on the network PID within a multiplex), and may be sent either broadcast-addressed (to all terminals in the network), multicast-addressed (to a group of terminals) or singlecast-addressed (to an individual terminal).

The controller 120 in FIG. 1 prefixes the virtual object message with a *configured_for_MAM* decoder condition in the message preamble. As a result, only terminals which are *configured_for_MAM* will process this message. This ensures that terminals which are not running a MAM capable firmware platform code will fail the decoder condition test, and will not acquire a VAT.

A terminal is considered to be in a MAM enabled state if it is *configured_for_MAM,* and has completely acquired the *home_VAT*.

Information derived from the Virtual Object message, including the VAT, is stored typically by the terminal in non-volatile memory (e.g., at the MAM 240), to preserve it through (warm) resets of the terminal.

The Virtual Object message can include the following significant fields:

"table_subtype" specifies that this Virtual Object message contains a VAT;

"VAT_ID" specifies an identifier for the VAT contained in this message. This ID may be the same as the *home_VAT_ID* from the *Virtual Application Config* message;

"sequence_number" specifies a version number for the VAT. If the *sequence_number* for the VAT included in this message is different from the *sequence_number* associated with the VAT, and the same *VAT_ID* is already present in the terminal, this implies that the VAT has changed;

"number_of_va_records" specifies how many VAT records are present in the VAT included in this message; and

"va_record" is an array of VAT records constituting the VAT. Each record identifies a virtual application. One of the records may identify the virtual application whose *default_application_ID* was given in the *Virtual Application Config* message.

Each record of the VAT can include the following significant fields:

"object_application_ID" contains a numeric identifier for the virtual application. The identifier should be unique among all va_records within a VAT;

"VCT_source_ID" is a list of identifiers of programming "sources" which are associated with the virtual application. Programming sources include any video, audio, or data "sources" that can be identified by a "source_ID", which is typically used to map a source name (e.g., ABC, HBO) to a virtual channel. The terminal may use these values to obtain a virtual channel to be tuned to before enabling the virtual application.

"VCT_application_ID" is a list of identifiers of "services" associated with the virtual application. The values and usage are the same as described for *VCT_source_ID* above;

"object_version" is a list of version numbers for each of the versions which can exist of a virtual application. The terminal will download the highest authorized version;

"virtual_application_tier" is a list of required authorization tiers for the virtual application, one per version. All versions of an application may have the same or different tiers. This specifies the authorization requirements for the versions of the virtual application; and

"virtual_name" is a multi-lingual text string of printable ASCII characters. The name can be used for on-screen displays at the terminal.

The *Tune Download Channel* Message is a sub-command of the Download Control message. A field *"tune_download_function_field" can* specify whether the message applies to a *"virtual_application"* or to a standard, non-MAM application.

The *Tune Download Channel* message for all virtual applications should contain the *configured_for_MAM* decoder condition in the message preamble. As a result, only terminals which are *configured_for_MAM* will process this message. This ensures that terminals which are not running a MAM-capable firmware platform

code will fail the decoder condition test, and will not acquire a virtual application.

If a virtual application is specified in the *Tune Download Channel* message, the virtual application is identified by the *obj_application_ID* field in the message. This virtual application then correlates to the one identified by the *object_application_ID* field in one of the records of the VAT (i.e., the *home_VAT*) maintained by the MAM 240 of FIG. 2. Moreover, the *obj_application_ID*, *tune_object_name* and *tune_object_version* in the *Tune Download Channel* message should correlate with the *application_ID*, *object_name* and *object_version*, respectively, in the Download message for the virtual application.

A *Tune Download Channel* message can be provided for a system-wide default virtual application. The *configured_for_MAM* decoder condition is not used for this default application. As a result, all terminals will always be able to acquire the system-wide default application.

The invention may also replace the use of Download Control messages. Since the MAM 240 has the information (via the VAT) about which applications should be enabled, disabled, purged, etc., the Downloader 230 can no longer directly act on the receipt of the Download Control sub-command message. As a result, if MAM is enabled, the "enable", "disable", "delete" and "purge" functions specified in

a Download Control message, for virtual applications, are ignored by the Downloader 230.

Also, if MAM is enabled, the "enable" function specified in a Download Control message for a virtual application causes the Downloader 230 to interrogate the MAM 240 to see if a particular application should indeed be enabled. The MAM responds back with instructions to enable or disable the virtual application.

The invention may also use a *"Virtual Channel Config Message"*. If MAM is enabled, the terminal will disregard the *turnon_VC_defined, turnon_VC, turnoff_VC_defined* and *turnoff_VC* fields specified by this message if the default virtual application has a defined *VCT_source_ID*. "VC" indicates a virtual channel. In this case, the terminal will tune to the channel associated with the *VCT_source_ID* given for the default virtual application.

The MAM feature requires the presence of versions of software, in the controller 120, and in a terminal 150, which are capable of executing the MAM functionality. The controller 120 should have a version of software which can create and send the new and modified messages to the terminals. The controller should be capable of providing one-way refreshes of specific configuration messages to terminals.

The controller should also provide multiple billing system authorization support for multiple applications.

All current terminals executing out of ROM code cannot be *configured_for_MAM*, because reserved entries and fields are used for implementing the MAM functionality.

Likewise, all terminals currently executing non-MAM capable software from flash memory should be downloaded with a version of software which is MAM capable that can acquire, understand and process the new and modified messages related to MAM functionality.

The invention implements a MAM while minimizing required changes to existing applications by providing a default virtual application, such as an electronic program guide (EPG), on a system-wide basis.

The *Tune Download Channel* message for the system-wide default virtual application is the only such message in the system that specifies the object as a virtual application, and does not require a *configured_for_MAM* decoder condition in the preamble of the message. Each Tune Download Channel message specifies an object to download. For standard applications, the message includes an application name and version as well as a channel where the application object is being transmitted. When a terminal receives this message, it tunes to the specified channel to acquire the object. For virtual applications, this message also includes the application_ID for the object in addition to the information included for a standard application.

As a result, any terminal that is not running a version of software that is MAM capable, will acquire

and enable the system-wide default application without recognizing it as a virtual application.

A terminal which is running a MAM capable version of firmware platform code can also acquire the system-wide default application. However, after acquiring the application, it will be treated as a virtual application to be managed by the MAM 240.

A MAM enabled terminal's default virtual application is downloaded into non-volatile memory at the terminal and also uses non-volatile memory for its settings so that the default is preserved even while another virtual application is enabled.

When MAM is enabled, the default virtual application, if present in the terminal, is typically enabled after any warm reset of the terminal, or when the terminal transitions from a "Terminal On" to a "Terminal Off" state.

FIG. 3 illustrates the creation of a local VAT for a terminal based on the terminal's authorized tiers in accordance with the present invention.

VAT data that is received by each MAM configured terminal in the network, shown generally at 300, includes a number of rows 310, 330, 350 and 370 of records (*va_records*). Records 312, 332, 352 and 372 contain the identifications (*object_application_ID*) of the first, second, third and fourth applications (i.e., Applications No. 1, No. 2, No. 3, and No. 4, respectively).

Records 314, 316 and 318 contain the version identifier (*object_version*) and required tier

(*virtual_application_tier*) for a first application. For example, record 314 has a version "1.0" and a required tier of "4", record 316 has a version "1.1" and a required tier of "5", and record 318 has a version "1.2" and a required tier of "7".

As an example, Application No 1. might be an e-mail capability, where version 1.0 is a basic version, version 1.1 is an enhanced version, and version 1.2 is a test version. Therefore, the invention allows a network operator to control which terminals can access which version of Application No. 1. Those who pay a small fee can access version 1.0, those who pay a larger fee can access version 1.1, and others can access version 1.2, e.g., to provide feedback to the network operator as to whether the version 1.2 works properly or provides desirable features.

For Application No. 2, record 334 has a version "2.0" and a required tier of "5", and record 336 has a version "2.1" and a required tier of "6".

For Application No. 3, record 354 has a version "7.5" and a required tier of "8".

For Application No. 4, record 374 has a version "1.0" and a required tier of "3", record 376 has a version "1.1" and a required tier of "4", record 378 has a version "2.0" and a required tier of "7", and record 379 has a version "2.1" and a required tier of "8".

Versions and tiers are always numeric values. The virtual_application_name is a text string that is associated with all versions of a virtual application

and is displayed on the dynamically built menu if one of the versions is authorized in the terminal.

The local VAT data that is authorized at the terminal, shown at 355, is assembled from the received VAT records 300 according to the tier (*virtual_application_tier*) with which the particular terminal is authorized. In the present example, it is assumed that the terminal's authorized rights include tiers "4" and "5". The local VAT data 355 and its tiers denote an authorization state of the terminal.

In this case, the matching tier requirements in the VAT records 300 are in records 316, 334 and 376. Accordingly, the terminal's local VAT 355 includes these records, e.g., stored in the VAT memory 260 of FIG. 2, as records 316', 334' and 376'. Note that when a terminal is authorized to receive two or more versions of the same application (e.g., records 314 and 316), it is generally desirable to select the most recent version (e.g., record 316). Generally, the required authorization state of an application is said to "correspond" to the authorization state of a terminal when the terminal's security processor has received a matching tier via an EMM.

If there is no matching tier for the terminal, e.g., as with Application No. 3, the terminal is not authorized to receive that application, and it is not downloaded.

It is possible to provide a required tier and/or authorized tier that allows a terminal to receive the highest version of all available applications.

Note that the VAT data 300 shown is only an example, and various numbers of applications, versions and tiers may be present in the VAT data. Additionally, the local VAT 355 may include all, some or none of the applications in the received home_VAT 300.

FIG. 4 illustrates an example dynamically-created menu of the available applications at a terminal in accordance with the present invention. The menu 400 presents the available authorized applications to the user, e.g., on a television screen. Note that the menu items correspond to the terminal's local VAT 355.

Each application may have a user-friendly textual data (*virtual_name*) associated with it. The messages provided to the terminal provide this textual data that describes the application. This data can be used on the display menu 400 of the terminal.

For example, referring to the example of FIG. 3, Application Nos. 1, 2 and 4 may have the textual data "E-mail", "Video-on-Demand, and "Web Browser", respectively.

Optionally, an additional textual string denotes the version of the application, or the corresponding service tier that the user has purchased. For example, for marketing purposes the versions may be denotes by bronze, silver or gold or the like.

Using the text associated with applications, the MAM 240 of FIG. 2 can dynamically construct a menu of authorized applications.

A user of a terminal can activate this menu and launch (e.g., start) one or more of the available applications by conventional interface techniques, e.g., using a key on a remote control. A particular key may be reserved on the remote control as a "MAM menu button".

Upon receiving VAT data from the controller 120, the MAM may build a menu consisting of a list of names of the authorized applications. This can be accomplished using known software techniques, for example. The menu may also include an "Exit" item to exit the menu. The menu may be implemented as a full or partial screen overlay with the audio muted. In addition, the front panel LEDs may be cleared while the menu is displayed.

A user can activate the display of the menu by pressing the "MAM menu button" on the remote control, and use the "up" or "down" keys on the remote control or on the front panel to scroll through the menu. A user can select an application for launch by pressing the "select" button on the remote control or the front panel of the terminal.

When an application is selected from the menu, the MAM 240 may inform the downloader 230 to enable the application, and the application is started (e.g., launched). The terminal may tune to a specified channel before enabling the application.

If the user selection is not authorized, or is not present in the VAT or in the downloaded objects directory, the MAM 240 will try to re-create the menu and re-display it. After several failed attempts to

launch a specific virtual application, the terminal will resort to some type of graceful recovery action.

The user can choose to not select any virtual application on the menu, as a result of which the MAM will re-enable the previously enabled (i.e. - prior to the menu being displayed) virtual application, or resort to a graceful recovery action that may enable the terminal's default virtual application.

FIG. 5 illustrates a method for providing multiple application management in a digital network in accordance with the present invention. A simplified overview of the method of the present invention is shown.

As shown at block 500, control data including terminal authorization rights, and MAM configuration data with an enable signal are sent to each terminal, or selected terminals, in a network. At block 510, application data (i.e., VAT data) with the version identifiers and authorization requirements (e.g., required tiers) for each application are transmitted to the terminals.

At block 520, the corresponding authorization rights and configuration data are stored at the terminals. As discussed, each terminal can have its own authorization state, which may be subsequently replaced or supplemented. At block 530, the terminal compares it authorization state to the authorization requirements for each version of the applications in the home VAT to build and store its local VAT. At block 540, the highest versions of the applications whose required authorization state corresponds to the

terminal's authorization state (i.e., of the versions in the local VAT) are downloaded. Lower versions (authorized or unauthorized) of the applications that are being replaced, if any, are erased at the terminal.

At block 550, the terminal continues to monitor the received control data and VAT data to determine if this data has been changed. If so, blocks 520, 530 and 540 are repeated. A change in the terminal's authorization state and/or the VAT data may or may not result in a new home or local VAT and downloading of new applications.

Accordingly, it can be seen that the present invention provides a method and apparatus for allowing a digital network to manage and utilize multiple applications in consumer terminals. Different terminals are authorized individually to download and access one or more of a number of available applications according to the authorization requirements of the application/version combination, and the authorized state of the terminal. Each terminal builds its own "local" VAT that identifies the application/versions for which it is authorized. The invention allows network operators to provide the different applications on a fee basis. Moreover, a customized on-screen menu for each terminal can be dynamically generated based on the terminal's local VAT data.

Although the invention has been described in connection with various specific embodiments, those skilled in the art will appreciate that numerous adaptations and modifications may be made thereto

without departing from the spirit and scope of the invention as set forth in the claims.

For example, while various syntax elements have been proposed herein, note that they are examples only, and any syntax may be used.

Moreover, while the invention was discussed in connection with a cable or satellite television broadband communication networks, it will be appreciated that other networks such as Digital Subscriber Loops (DSLs), local area networks (LANs), metropolitan area networks (MANs), wide area networks (WANs), internets, intranets, and the Internet, or combinations thereof, may be used.

What is claimed is:

1. A method for managing multiple applications in a digital network having a headend that broadcasts programming services to a terminal population via a communication channel, comprising the steps of:

communicating configuration data from the headend to terminals in the terminal population via the communication channel to configure the terminals to receive application data and control data; wherein:

said application data is provided for a plurality of applications, and defines an identifier, a version, and a required authorization state for each of the applications; and

said control data defines respective authorization states for the terminals; and

communicating said application data and said control data from the headend to the terminals via the communication channel to enable the terminals to download and access the versions of the applications for which the required authorization state thereof corresponds to the terminal's authorization state.

2. The method of claim 1, wherein:

the required authorization state for each terminal corresponds to one of a plurality of available service level tiers offered by the headend.

3. The method of claim 2, wherein:

the available service level tiers are offered by the headend upon payment of corresponding fees by users of the terminals.

4. The method of claim 1, wherein:

the control data is communicated from the headend to the terminals by addressing individual ones of the terminals.

5. The method of claim 1, wherein:

the control data is communicated from the headend to the terminals by addressing a group of the terminals.

6. The method of claim 1, wherein:

said control data defines a global authorized tier for enabling at least one of the terminals to access all of the applications.

7. The method of claim 1, comprising the further step of:

providing a billing system for billing specific ones of the terminals according to their authorization state.

8. The method of claim 1, wherein the applications include at least one of:

electronic program guide, Internet browser, video-on-demand, audio-on-demand, mail service, telephony service, stock prices, weather data, travel information, games, gambling, banking, shopping, and voting.

9. The method of claim 1, wherein

the application data is communicated via the communication channel in transport packets in a separate band from a band in which the programming services are broadcast.

10. The method of claim 1, comprising the further step of:

generating an on-screen menu at the terminals using the application data thereof according to the versions of the applications the terminals are authorized to access.

11. The method of claim 1, wherein:

when at least two versions of one of the applications have required authorization states that correspond to the authorization states of one of the terminals, the highest one of the versions is downloaded and accessed by the one of the terminals.

12. The method of claim 1, wherein:

the configuration data identifies a default application for the terminals.

13. The method of claim 1, wherein:

the configuration data provides information to the terminals for downloading the application data from the communication channel.

14. The method of claim 1, wherein:

the digital network is a broadband television communication network.

15. The method of claim 1, wherein:

the control data specifies a particular programming service that is to be tuned by the terminals when a corresponding application is launched thereat.

16. The method of claim 1, comprising the further step of:

storing the control and configuration data at the terminals for subsequent use in downloading and accessing the versions of the applications for which the required authorization state thereof corresponds to the terminal's authorization state

17. The method of claim 1, wherein:

the application data includes revision data that allows the terminals to determine when revised application data is being communicated thereto; and

the terminals are responsive to the revision data for determining whether the versions of the

18. The method of claim 1, wherein:

when the versions of the applications for which the terminals are authorized to access has changed, the terminals delete data stored thereat corresponding to a replaced version of the authorized application, and store data thereat corresponding to a new version of the authorized application.

19. The method of claim 1, wherein:

the control data causes the terminals to download the versions of the applications for which the terminals are authorized to access, while rejecting the versions of the applications for which the terminals are not authorized to access.

20. The method of claim 1, wherein:

the configuration data causes the terminals to allocate an amount of memory thereat for storing the versions of the applications for which the terminals are authorized to access.

21. The method of claim 1, wherein:

said identifier, version, and required authorization state for each of the applications are provided to the terminals in a common virtual application table; and

each terminal builds it own local virtual application table from the common virtual application table in accordance with the respective authorization state to define the version(s) of the application(s) for which the required authorization state corresponds to the terminal's authorization state.

22. The method of claim 1, wherein:

if the authorization state of one of the terminals is downgraded, at least one application already stored thereat but no longer authorized is deleted.

23. A method for managing multiple applications in a terminal of a digital network, said digital network having a headend that broadcasts programming services to a terminal population, including said terminal, via a communication channel, comprising the steps of:

receiving configuration data from the headend at the terminal in the terminal population via the communication channel to configure the terminal to receive application data and control data; wherein:

said application data is provided for a plurality of applications, and defines an identifier, a version, and a required authorization state for each of the applications; and

said control data defines respective authorization states for the terminal; and

receiving said application data and said control data from the headend at the terminal via the communication channel for use in downloading and accessing the version(s) of the application(s) for which the required authorization state thereof corresponds to the terminal's authorization state.

24. An apparatus for managing multiple applications in a digital network having a headend that broadcasts programming services to a terminal population via a communication channel, comprising:

means for communicating configuration data from the headend to the terminals in the terminal population via the communication channel to configure the terminals to receive application data and control data; wherein:

said application data is provided for a plurality of applications, and defines an identifier, a version, and a required authorization state for each of the applications; and

said control data defines respective authorization states for the terminals; and

means for communicating said application data and said control data from the headend to the terminals via the communication channel to enable the terminals to download and access the versions of the applications for which the required authorization state thereof corresponds to the terminal's authorization state.

25. A terminal for managing multiple applications, said terminal being provided in a digital network having a headend that broadcasts programming services to a terminal population, including said terminal, via a communication channel, comprising:

means for receiving configuration data from the headend via the communication channel to configure the terminal to receive application data and control data; wherein:

said application data is provided for a plurality of applications, and defines an identifier, a version, and a required authorization state for each of the applications; and

said control data defines respective authorization states for the terminal; and

means for receiving said application data and said control data from the headend at the terminal via the communication channel for use in downloading and accessing the version(s) of the application(s) for which the required authorization state thereof corresponds to the terminal's authorization state.

AMENDED CLAIMS
[received by the International Bureau on 15 August 2000 (15.08.00);
original claim 15 amended; remaining claims unchanged (1 page)]

13. The method of claim 1, wherein:
the configuration data provides information to the terminals for downloading the application data from the communication channel.

14. The method of claim 1, wherein:
the digital network is a broadband television communication network.

15. The method of claim 1, comprising the further step of:
specifying a particular programming service that is to be tuned by the terminals when a corresponding application is launched thereat.

16. The method of claim 1, comprising the further step of:
storing the control and configuration data at the terminals for subsequent use in downloading and accessing the versions of the applications for which the required authorization state thereof corresponds to the terminal's authorization state

17. The method of claim 1, wherein:
the application data includes revision data that allows the terminals to determine when revised application data is being communicated thereto; and
the terminals are responsive to the revision data for determining whether the versions of the

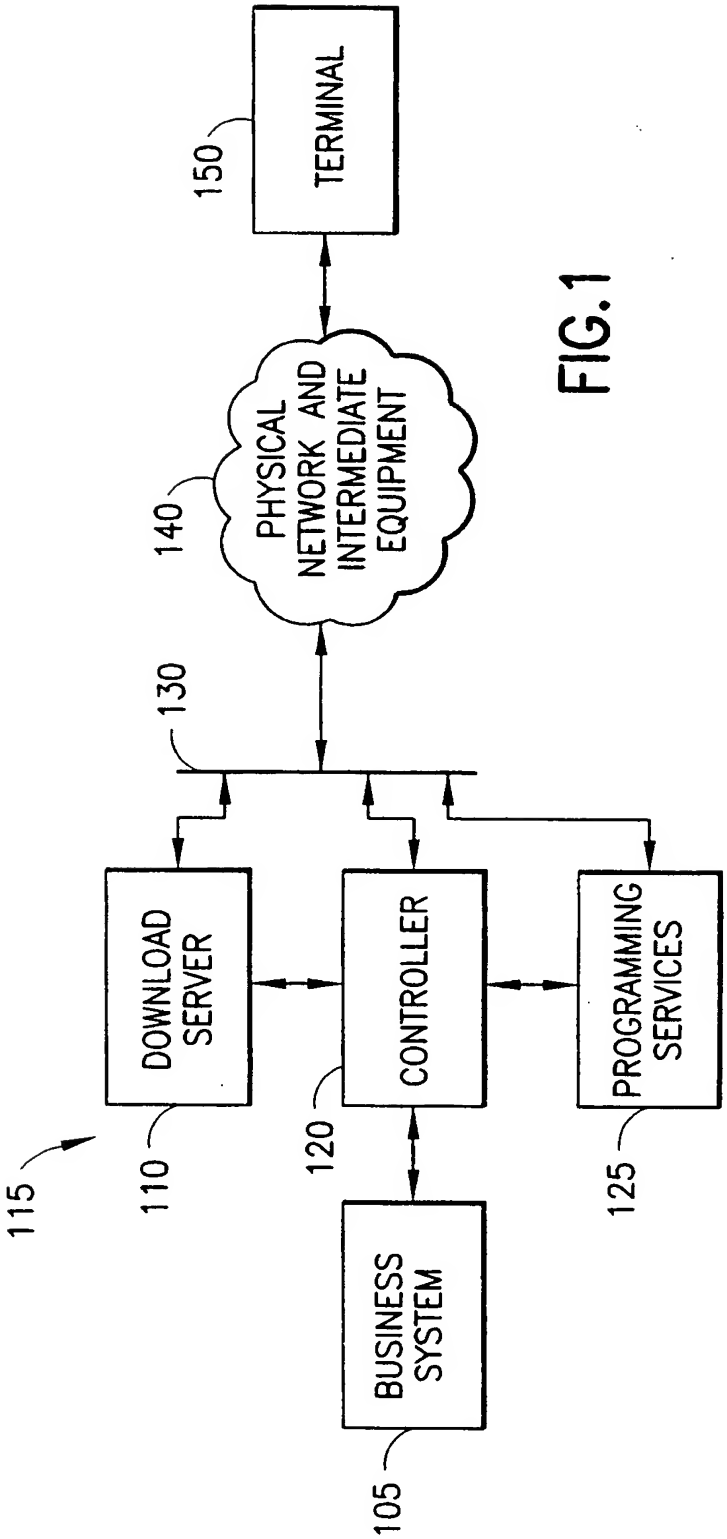

TERMINAL
150
PHYSICAL NETWORK AND INTERMEDIATE EQUIPMENT
140
130
DOWNLOAD SERVER
CONTROLLER
PROGRAMMING SERVICES
BUSINESS SYSTEM
115
110
120
125
105

FIG.1

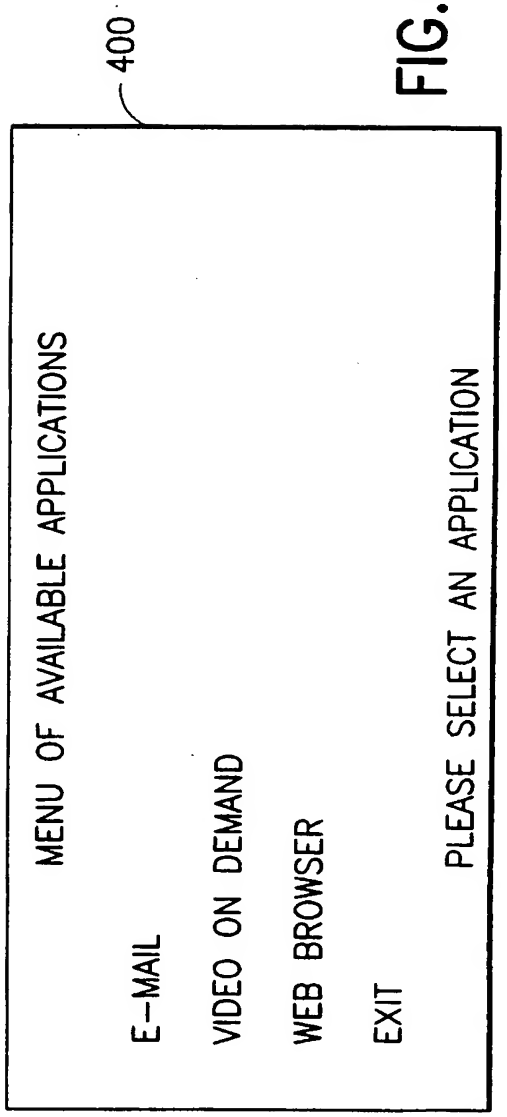

400
MENU OF AVAILABLE APPLICATIONS
E-MAIL
VIDEO ON DEMAND
WEB BROWSER
EXIT
PLEASE SELECT AN APPLICATION

FIG.4

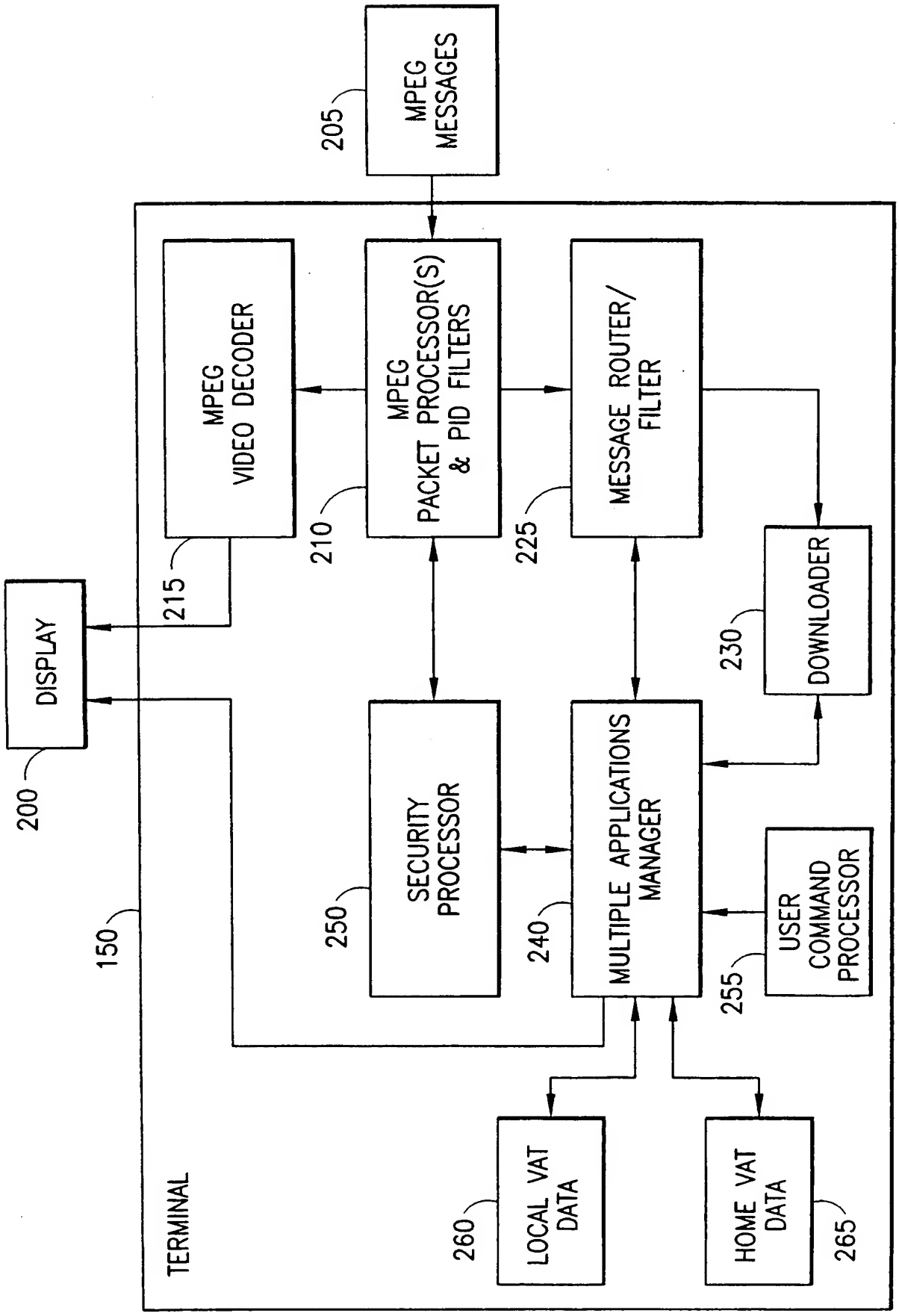

MPEG MESSAGES
205
MPEG VIDEO DECODER
MPEG PACKET PROCESSOR(S) & PID FILTERS
MESSAGE ROUTER/ FILTER
DISPLAY
215
210
225
230
DOWNLOADER
200
SECURITY PROCESSOR
MULTIPLE APPLICATIONS MANAGER
USER COMMAND PROCESSOR
250
240
255
150
TERMINAL
LOCAL VAT DATA
HOME VAT DATA
260
265

FIG.2

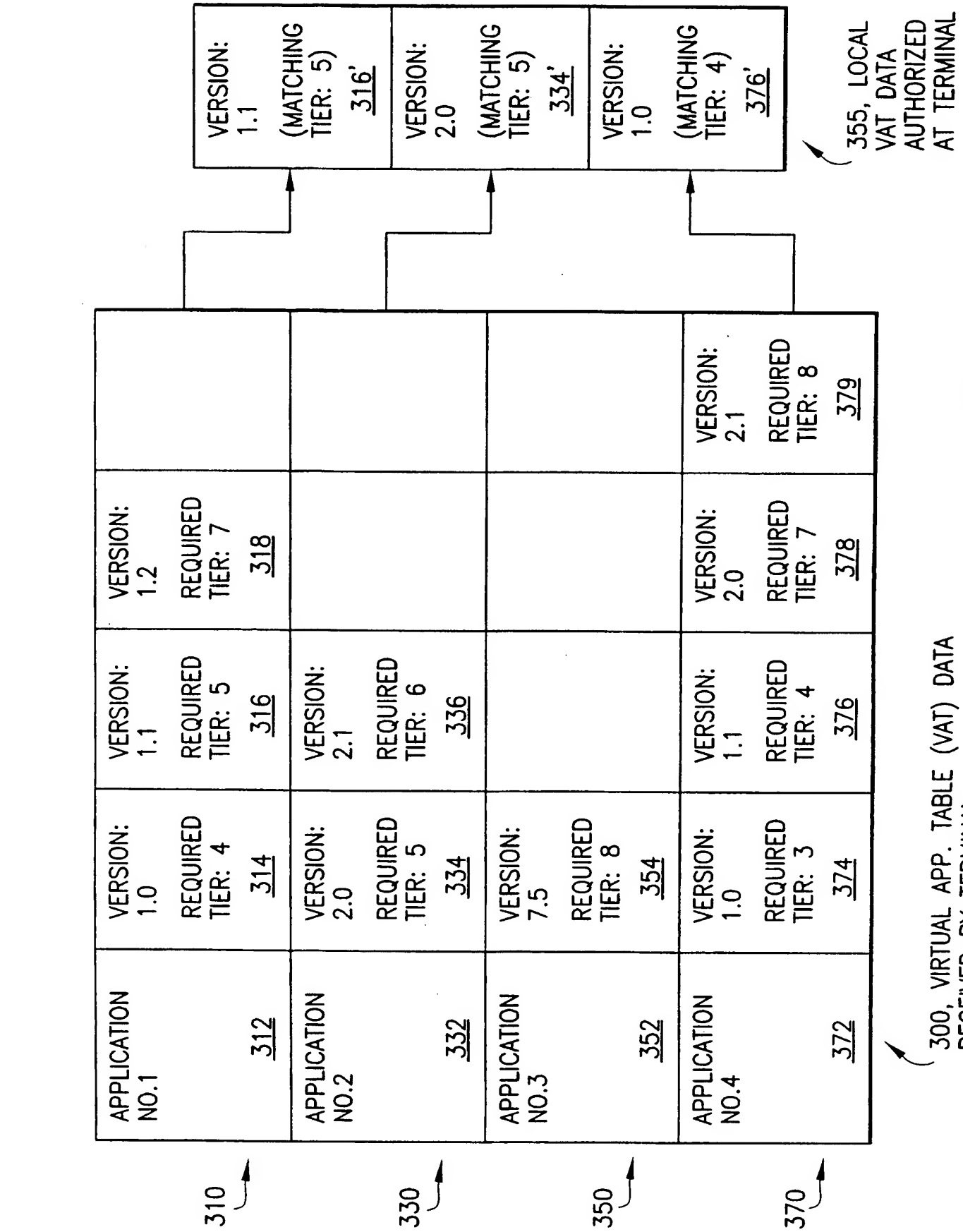

APPLICATION NO.1 312
VERSION: 1.0 REQUIRED TIER: 4 314
VERSION: 1.1 REQUIRED TIER: 5 316
VERSION: 1.2 REQUIRED TIER: 7 318
APPLICATION NO.2 332
VERSION: 2.0 REQUIRED TIER: 5 334
VERSION: 2.1 REQUIRED TIER: 6 336
APPLICATION NO.3 352
VERSION: 7.5 REQUIRED TIER: 8 354
APPLICATION NO.4 372
VERSION: 1.0 REQUIRED TIER: 3 374
VERSION: 1.1 REQUIRED TIER: 4 376
VERSION: 2.0 REQUIRED TIER: 7 378
VERSION: 2.1 REQUIRED TIER: 8 379
VERSION: 1.1 (MATCHING TIER: 5) 316'
VERSION: 2.0 (MATCHING TIER: 5) 334'
VERSION: 1.0 (MATCHING TIER: 4) 376'
310
330
350
370
300, VIRTUAL APP. TABLE (VAT) DATA RECEIVED BY TERMINAL
355, LOCAL VAT DATA AUTHORIZED AT TERMINAL

FIG.3

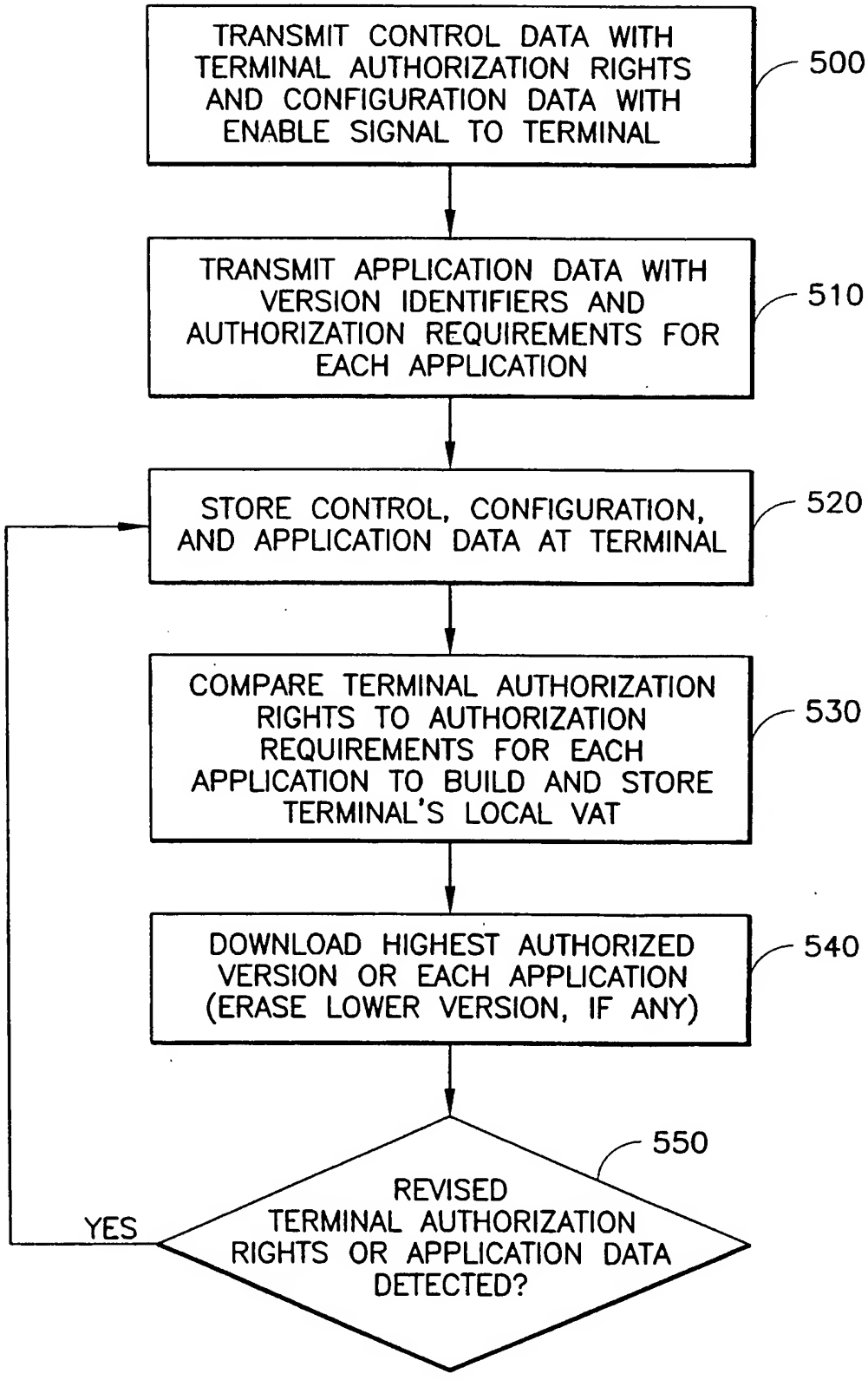

TRANSMIT CONTROL DATA WITH TERMINAL AUTHORIZATION RIGHTS AND CONFIGURATION DATA WITH ENABLE SIGNAL TO TERMINAL
500
TRANSMIT APPLICATION DATA WITH VERSION IDENTIFIERS AND AUTHORIZATION REQUIREMENTS FOR EACH APPLICATION
510
STORE CONTROL, CONFIGURATION, AND APPLICATION DATA AT TERMINAL
520
COMPARE TERMINAL AUTHORIZATION RIGHTS TO AUTHORIZATION REQUIREMENTS FOR EACH APPLICATION TO BUILD AND STORE TERMINAL'S LOCAL VAT
530
DOWNLOAD HIGHEST AUTHORIZED VERSION OR EACH APPLICATION (ERASE LOWER VERSION, IF ANY)
540
REVISED TERMINAL AUTHORIZATION RIGHTS OR APPLICATION DATA DETECTED?
550
YES

FIG.5

INTERNATIONAL SEARCH REPORT

Inte onal Application No
PCT/US 99/24745

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 H04N7/16

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 H04N

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)
EPO-Internal, WPI Data, PAJ

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	EP 0 399 200 A (GEN INSTRUMENT CORP) 28 November 1990 (1990-11-28)	1,4,7-9, 12,13, 18,23-25
Y	column 1, line 7 - line 21 column 4, line 11 - line 17 column 6, line 37 - line 43 column 7, line 9 - line 20 column 8, line 48 - line 55 column 9, line 3 - line 17 column 10, line 7 - line 21 column 14, line 40 - line 50 column 16, line 55 -column 17, line 4	10,16, 20,22

-/--

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
23 June 2000	30/06/2000

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Sindic, G

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Inte onal Application No
PCT/US 99/24745

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication,where appropriate, of the relevant passages	Relevant to claim No.
X	US 5 654 746 A (MCMULLAN JR JAY C ET AL) 5 August 1997 (1997-08-05) column 7, line 62 -column 8, line 3 column 9, line 49 -column 10, line 8 column 10, line 48 - line 57 column 11, line 45 -column 12, line 6	1,4,8, 10,16, 21,23-25
Y A	US 5 951 639 A (MACINNIS ALEXANDER G) 14 September 1999 (1999-09-14) column 3, line 54 - line 64 column 4, line 20 - line 41 column 5, line 13 - line 23 column 6, line 5 - line 12 column 7, line 62 -column 8, line 31	10,16, 20,22 15
A	US 5 734 589 A (HUDSON JR HENRY G ET AL) 31 March 1998 (1998-03-31) column 4, line 39 - line 48 column 15, line 11 - line 27 column 21, line 39 - line 55 column 22, line 10 - line 30 column 24, line 39 - line 52 column 27, line 16 - line 30 column 28, line 8 - line 39 column 31, line 58 -column 32, line 7	1,4,8, 10,14, 15,23-25

Form PCT/ISA/210 (continuation of second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Information on patent family members

Inte ional Application No
PCT/US 99/24745

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 0399200 A	28-11-1990	US 5003591 A	26-03-1991
		AT 154741 T	15-07-1997
		AU 617279 B	21-11-1991
		AU 5481990 A	29-11-1990
		CA 2013982 A	25-11-1990
		DE 69030933 D	24-07-1997
		DE 69030933 T	08-01-1998
		DK 399200 T	29-12-1997
		EP 0732850 A	18-09-1996
		HK 1008411 A	07-05-1999
		IE 80417 B	01-07-1998
		JP 3021184 A	29-01-1991
US 5654746 A	05-08-1997	AU 688141 B	05-03-1998
		AU 3640695 A	19-06-1996
		BR 9509857 A	30-12-1997
		CA 2206234 A	06-06-1996
		EP 0795253 A	17-09-1997
		JP 10510408 T	06-10-1998
		WO 9617475 A	06-06-1996
		US 6029046 A	22-02-2000
US 5951639 A	14-09-1999	AU 1693597 A	02-09-1997
		EP 0880857 A	02-12-1998
		WO 9730549 A	21-08-1997
US 5734589 A	31-03-1998	US 5666293 A	09-09-1997
		US 5768539 A	16-06-1998
		US 5978855 A	02-11-1999